

# AWS SNOWBALL EDGE: AN INTRODUCTION



There's an old saying in the IT community that goes something like:

*"Never underestimate the bandwidth of a truckload of hard drives barreling down the highway."*

This interesting imagery comes from the ever-growing competition between internet speed and data loads. Long story short, the more data in need of transfer, the longer it will take via the internet. After all, the internet and the cloud are extremely useful in many contexts—but transmitting terabytes of data isn't one of them.

There eventually comes a point where it's faster to ship physical hard drives than to try downloading the immense amounts of information they contain. The [Amazon Web Services \(AWS\)](#) Snowball Edge fits such situations by providing a simple, safe, and scalable physical device that can conform to the needs of practically any project.

The AWS Snowball Edge addresses bulk data transfer in the age of [cloud computing](#) but pushes it a step further. It takes the original AWS Snowball devices and adds:

- Cutting-edge computing functionality
- On-board storage for select AWS capabilities.

This combination provides advanced "edge computing" for those times when you need temporary solutions to problems, such as remote geographic location or a lack of high-speed internet connection.

# The original AWS Snowball device

The AWS Snowball Edge is a type of the broader [Snowball](#) device classification. AWS Snowballs in general offer petabyte-scale solutions for hefty data migration both into and out of the AWS cloud. Common uses include the transfer of:

- Analytics data
- Video libraries
- Backups
- Genomics data
- Other large-scale archives

Transferring such substantial amounts of data via bandwidth often leads to problems like:

- Long transfer times
- High costs
- Concerns about security

The classic AWS Snowball circumvents these issues through the use of a physical “management console” that ships directly to the user. The console is a sturdy little appliance that acts as a built-in shipping container with an automatic E-Ink shipping label. Upon arrival, an AWS Snowball console must establish a connection to your local network after you download and run a Snowball Client. The Client is used to select and encrypt the data you want to transfer to the console at high speeds.

Transferring directly to a Snowball device avoids the use of costly and sluggish bandwidth. For example, transferring 100 terabytes via high-speed 100 Mbps internet can take several months. Alternatively, the same transfer using two Snowball consoles takes less than a week.

Such a reduction in bandwidth usage not only reduces the wait time, but also cuts costs significantly compared to the potentially thousands of dollars required for long-term, high-speed internet. The AWS Snowball Edge applies these same ideas, but with some added edge computing capabilities for particularly demanding situations.

## Adding edge computing

Although most people and companies own personal computers and other hardware, they typically use them via centralized services in the cloud. The entire goal of the original AWS Snowball is to transfer data to the cloud, but providers are reaching the limits of improving cloud capabilities.

Now that these companies have centralized the cloud on a vast scale, the existing small limitations have become more apparent. For instance, the limit of the speed of light causes short but perceivable delays in everything from opening your browser to gaming online as various signals ping across the globe.

Overcoming these shortfalls has led to “[edge computing](#)” and the “[intelligent edge](#)”. In this context, “edge” refers both to:

- The figurative edge of the cloud computing era
- The literal geographic edge, where it's not possible or practical to rely on the cloud due to location

In addition to transferring data onto the cloud, an edge computing device can process it locally, undertaking large computing workloads. Conducting edge computing at or near the sources of data rather than transmitting it across the planet via the cloud significantly cuts down on delays. Essentially, the cloud comes to you—you no longer have to be in the right place with the right equipment.

Some examples of current edge computing applications include:

- Encrypting and storing information on devices such as smartphones to offload reliance on the cloud
- Loading and saving locally onto websites before syncing periodically to the cloud,
- Running [artificial intelligence](#) on devices to make them smart about how to use the cloud.

The AWS Snowball Edge employs the technology by providing enhanced local computation power and skipping the Snowball Client download in favor of automatically encrypting data on the devices. Each of these applications has many benefits. Users get faster and more reliable services. Companies reap the benefits of flexible [hybrid cloud computing](#).

## AWS Snowball Edge benefits

All Snowball devices allow the transfer of bulk data in and out of Amazon's Simple Storage Service (S3), use the same application programming interface (API) for job management, and have similar consoles.

The AWS Snowball Edge contributes additional edge computing features and improved local storage. Specifically, it allows:

- Clustering of devices
- Local compute with [AWS Lambda](#)
- Use with AWS IoT Greengrass and Amazon Elastic Compute Cloud (EC2) compute instances

In terms of storage capacity:

- The AWS Snowball device only offers 50 or 80 TB of capacity with 42 or 72 TB of that being usable, respectively.
- The AWS Snowball Edge has 100 TB with 83 TB usable. (There is also a 100 TB clustered option that offers 45 usable TB per node.)

Clustering offers the ability to achieve near-perfect data durability across 5-10 devices while growing or shrinking storage locally on demand.

While the idea and execution of both devices are quite similar, the AWS Snowball Edge doesn't depend on downloading a client to perform tasks. If using an Amazon S3 adapter for Snowball, then the AWS Snowball Edge device comes with the client pre-installed. It then encrypts data automatically upon transfer to the AWS Snowball Edge device, whereas using a downloaded Snowball Client encrypts the data on your workstation before data transfer to the Snowball device.

## Additional options & features

The AWS Snowball Edge comes with three device configuration options to choose from:

- **Storage Optimized**, which provides Amazon S3-compatible object storage, block storage, and 24 vCPUs, is a good option for both large-scale data transfer and local storage.
- **Compute Optimized** is likely best for advanced machine learning and full-motion video analysis, thanks to 52 vCPUs and object and block storage.
- **An optional onboard GPU** is an add-on to the compute-optimized version that enhances these capabilities.

In addition to Amazon S3 compatibility, the Amazon EC2 endpoint supports a subset of API actions for running and computing instances on the AWS Snowball Edge. You can also trigger Lambda functions in response to events, allowing you to run code without any administration.

## Using AWS Snowball Edge

Pairing device options with AWS Lambda functions and Amazon EC2 instance types allows users to perform development and testing in AWS before deploying applications on devices in any location. Common uses with these various options include:

- Data migration and transport
- Machine learning
- Image collation
- [IoT sensor](#) stream capture

For example, some entities who have already taken advantage of the AWS Snowball Edge technology include [Boeing and the Oregon State University research team](#).

Boeing put it to use while flying image-capturing drones in remote areas. The AWS Snowball Edge allowed them to pre-process the data on the spot before transferring it to the cloud for analysis. Similarly, the OSU team needed to collect data on a distant research vessel, analyze it, and transfer it to the cloud after arriving back home.

If you're thinking about making use of the AWS Snowball edge technology, do some research into the steps you should take. You'll need to create an AWS account that includes administrator-level AWS Identity and Access Management permissions. The files and folders you want to transfer also need to follow object key naming conventions for Amazon S3. Exact steps depend on whether you wish to import data, export data, or use compute instances, so [read up on the details](#) to ensure you get the job done right.

## Related reading

- [BMC Multi-Cloud Blog](#)
- [AWS Guide](#), a series of articles & tutorials
- [Edge AI: Edge Artificial Intelligence Explained](#)
- [The Empowered Edge: An Introduction](#)
- [Structured vs Unstructured Data: A Shift in Privacy](#)